

APPLICATION  
FOR  
UNITED STATES LETTERS PATENT

TITLE: ADAPTIVE EQUALIZER  
APPLICANT: STANLEY K. LING AND HIROSHI TAKATORI

CERTIFICATE OF MAILING BY EXPRESS MAIL

Express Mail Label No. ET371085841US

I hereby certify that this correspondence is being deposited with the United States Postal Service as Express Mail Post Office to Addressee with sufficient postage on the date indicated below and is addressed to the Commissioner for Patents, Washington, D.C. 20231

8-15-01  
Date of Deposit

Signature

Gil Vargas  
Typed or Printed Name of Person Signing Certificate

## **ADAPTIVE EQUALIZER**

### **TECHNICAL FIELD**

This invention relates to an adaptive equalizer for data  
5 communications.

### **BACKGROUND**

Information may be transmitted from a transmitter to a  
receiver over a communication channel, such as a satellite  
10 link, a fiber optic cable, or a copper cable. The information  
may include, e.g., analog voice information from a telephone  
conversation or digital data that is transmitted between two  
computers. Analog information is often transformed into a  
digital format before it is transmitted over a channel.

15 The transmitted information may be encoded into a  
sequence of symbols selected from a set of pre-defined  
symbols, known as an alphabet. Each of the symbols is  
represented by an electronic pulse, which is transmitted over  
the communication channel to a remote location. The sequence  
20 of pulses forms a signal that is received by the receiver.  
The receiver retrieves the symbols from the signal and decodes  
them to recover the transmitted information.

Communication channels typically distort the pulses as they are transmitted through the channels. The channels may add noise to the pulses. Certain types of additive noise, known as white noise, are evenly distributed at all frequencies. Thermal noise from copper wires is typically a source of white noise. Other types of additive noise, known as colored noise, may be concentrated at certain frequencies. Signals induced on a wire by an adjacent wire, which is known as crosstalk, are a typical source of colored noise.

The channel may also distort the amplitude or phase of the transmitted signals. As a result of this distortion, the pulses representing the symbols may be corrupted with information from other pulses in the sequence. The corruption is referred to as inter-symbol interference (ISI). There are two kinds of ISI. A pulse representing a particular symbol may be corrupted with information from an earlier pulse in the sequence. This is known as post-cursor ISI. Alternatively, the pulse may be corrupted with information from a future pulse in the sequence. This is known as pre-cursor ISI.

The receiver typically has a signal detector to detect symbols received from the channel. The detector may, for example, be in the form of a simple threshold detector or a maximum likelihood sequence decoder. The detector is

typically optimized to detect symbols that have only been distorted by additive white Gaussian noise (AWGN). Consequently, colored noise and/or inter-symbol interference may cause errors when the symbols are detected.

5

#### DESCRIPTION OF THE DRAWINGS

Fig. 1 is a block diagram of an adaptive decision feedback equalizer (DFE).

Fig. 2 is a block diagram of a feedforward filter in the adaptive DFE.

Fig. 3 is a view of a computer system on which coefficients for the feedforward filter may be determined.

#### DESCRIPTION

Fig. 1 shows a block diagram of a channel (h(t)) 10 and an adaptive DFE 12. A signal  $A_k$ , comprised of a sequence of symbols, is transmitted over channel 10 to adaptive DFE 12. The resulting signal  $s(t)$  contains both pre-cursor and post-cursor ISI. Crosstalk, comprised of colored noise  $n(t)$ , is added to  $s(t)$  during transmission. Colored noise (also referred to as "colored Gaussian noise") is noise that varies over a range of frequencies. By contrast, white noise (also

referred to as "white Gaussian noise") is noise that is substantially the same over a range of frequencies.

The combination of  $s(t)$  and  $n(t)$ , namely  $r(t)$ , is sampled by adaptive DFE 12 at a sampling rate of  $t_0 + kT$  to produce a discrete time signal  $R_k$ , where  $t_0$  accounts for the channel delay and sampler phase.  $R_k$  is applied to feedforward filter (FFF) 14, which produces an output signal  $U_k$ . In this embodiment, FFF 14 is finite impulse response (FIR) filter that contains a number  $N_{fff}$  ( $N_{fff} \geq 1$ ) of taps. Among the  $N_{fff}$  taps is a main tap,  $M$  ( $1 \leq M \leq N_{fff}$ ), which is roughly at the center of the taps. Fig. 2 shows a block diagram of FFF 14.

For the purposes of this description, the taps of FFF 14 are divided into two sets, taps 16 (comprised of the taps from 1 to  $M$ ) and taps 18 (comprised of the taps from  $M+1$  to  $N_{fff}$ ). Taps 16 contain coefficients  $b_k^i$  (for  $i=1$  to  $M$ ) that are used by FFF 14 to reduce pre-cursor ISI in  $R_k$ . Taps 18 contain different coefficients  $b_k^i$  (for  $i=M+1$  to  $N_{fff}$ ) that are used by FFF 14 to whiten the noise in  $R_k$ , where the noise is  $n(t)$  (sampled,  $n_k$ ). Taps 18 do not necessarily reduce post-cursor ISI in  $R_k$ . To whiten the noise, the coefficients on taps 18 reduce the differences in noise  $n_k$  over the range of frequencies defined by the sampler's Nyquist frequency,  $kT/2$ .

A description of how the coefficients,  $b_k^i$ , are generated is provided below.

The output of FFF 14, namely  $U_k$ , contains reduced pre-cursor ISI and whitened noise. FBF 20 reduces post-cursor ISI in the output of adaptive DFE 12. In this embodiment, FBF 20 is an FIR filter that contains a number  $N_{fbf}$  ( $N_{fbf} \geq 1$ ) of taps. The hardware shown in the block diagram of Fig. 2 may also be used to construct FBF 20. Taps in FBF 20 contain coefficients  $d_k^i$  (for  $i=1$  to  $N_{fbf}$ ) that are used by FBF 20 to reduce post-cursor ISI in  $U_k$ . A description of how the coefficients,  $d_k^i$ , are generated is provided below.

The feedback loop 22 that includes FBF 20 works as follows. The output of FBF 20, namely  $Q_k$ , is provided to circuit junction 24, which adds  $Q_k$  to  $U_k$ . The resulting signal  $Y_k$  contains relatively little post-cursor ISI (since the post-cursor ISI,  $Q_k$ , is removed from  $U_k$ ). The signal  $Y_k$  is applied to slicer 26, which makes symbol (e.g., bit) decisions based on the content of  $Y_k$ . For example, slicer 26 may determine that a signal having a value that is greater than "0.0" constitutes a "1" bit and that a signal having a value that is less than "0.0" constitutes a "0" bit.

If correct bit decisions are made by slicer 26, the resulting signal  $X_k$  will be a replica of  $A_k$ . That is,  $X_k$  will

be the same as the original data signal  $A_k$ , meaning that  $X_k$  has no crosstalk noise, pre-cursor ISI, or post-cursor ISI. The difference, or error, between the ideal signal and the received signal is determined by taking the difference of  $X_k$  and  $Y_k$  using circuit junction 28. As described in more detail below, the error,  $E_k$ , is used to adaptively determine the coefficients  $b_k^i$  for FFF 14 and the coefficients  $d_k^i$  for FBF 20.

The output  $X_k$  of slicer 26 is also applied to feedback loop 30. Feedback loop 30 contains a channel estimator (CE) 32. CE 32 filters  $X_k$  to estimate the ISI added by channel 10. In this embodiment, CE 32 is an FIR filter with z-domain response,  $G_k(z)$ . CE 32 performs the appropriate filtering on  $X_k$ , resulting in signal  $T_k$ , which corresponds to an estimate of the signal component of  $R_k$  delayed in time. The amount of the delay is equal to the delay  $(M+L)$ , where delay  $M$  corresponds to the delay through FFF 14, circuit junction 24, and slicer 26, among other components (not shown), and delay  $L$  corresponds to the number of precursor samples to be replicated by CE block 32.

To obtain an estimate,  $V$ , of the noise  $(n_k)$ , adaptive DFE 12 obtains the difference between  $T_k$  and a delayed version of  $R_k$ . Delay circuit 34 ( $Z^{(M+L)}$ ) delays  $R_k$  by the delay through adaptive DFE 12, namely  $M$ , and an amount  $L$ , which corresponds

to the number of precursor samples to be replicated by CE 32.  $T_k$  is the estimate of the signal component of  $R_{k-M-L}$ . Circuit junction 38 subtracts  $T_k$  from the delayed version of  $R_k$ , namely  $R_{k-M-L}$ . The resulting signal,  $V_{k-M-L}$ , is an estimate of the  
5 noise  $n_k$ .  $V_{k-M-L}$  is referred to as an "estimate", rather than a measurement of the actual noise, because the value of  $T_k$  may not exactly match the signal component of  $R_{k-M-L}$ .

Described below are examples of ways of generating filter tap coefficients for FFF 14 and FBF 20. In this regard, prior art processes for generating pre-cursor and post-cursor tap coefficients for FFF 14 generated both types of coefficients in the same manner. Thus, in the prior art, the post-cursor taps of the FFF are designed both to whiten noise and to cancel post-cursor ISI. In a channel that introduces severe ISI and colored noise to a signal, such post-cursor taps generally prevent the FFF from adapting to an optimal weighted matched filter (WMF) solution.

By contrast, in adaptive DFE 12 described herein, the post-cursor taps of FFF 14 whiten noise, but are not designed to cancel post-cursor ISI (although some incidental cancellation of post-cursor ISI may occur). Consequently, the pre-cursor and post-cursor taps of FFF 14 are adapted using different least mean square (LMS) processes.



Referring to Fig. 1, adaptive DFE 12, and the equations that follow, are specified for a T-spaced adaptive DFE signal receiver utilizing a real, baseband modulation. To reiterate, in Fig. 1

$A_k$ : represents transmit data symbols at time index  $k$   
 $s(t)$ : represents continuous time output of the channel  $h(t)$  at time  $t$   
 $n(t)$ : represents additive channel noise at time  $t$   
 $r(t)$ : represents continuous time receiver input at time  $t$   
 $R_k$ : represents sampled receiver input at time index  $k$   
 $U_k$ : represents the output of FFF 14  $B_k(z)$  at time index  $k$   
 $Q_k$ : represents the output of FBF 20  $D_k(z)$  at time index  $k$   
 $Y_k$ : represents the equalized signal at time index  $k$   
 $X_k$ : represents the recovered symbol at time index  $k$   
 $E_k$ : represents the decision error  $(X_k - Y_k)$  at time index  $k$   
 $h(t)$ : represents the continuous-time channel impulse response. The channel is modeled as a continuous-time impulse response,  $h(t)$ , with the signal peak at time  $t_0$ .  
 $B_k(z)$ : represents the z-domain response of FFF 14 at time index  $k$   
 $D_k(z)$ : represents the z-domain response of FBF 20 at time index  $k$   
 $V_k$ : represents the estimated noise at time index  $k$   
 $T_k$ : represents the output of the CE  $G_k(z)$  at time index  $k$   
 $G_k(z)$ : represents the CE z-domain response at time index  $k$

To determine the coefficients, define  $A_k$  to be a pulse amplitude modulation (PAM) sequence. The received signal  $r(t)$  thus constitutes the superposition of the impulse response of the channel  $h(t)$  and each transmitted symbol and the additive noise  $n(t)$ . The noise may be either colored or white Gaussian noise. The received signal,  $r(t)$ , is given as:

$$r(t) = \sum_i A_i h(t - iT) + n(t) .$$

The received signal is sampled at instant  $kT + t_0$  to generate  $R_k$ , where  $t_0$  accounts for the channel delay and sampler phase:

$$R_k = A_k h(t_0) + \sum_{i \neq k} A_i h(t_0 + kT - iT) + n(t_0 + kT) .$$

In this embodiment, FFF 14 is an FIR filter with z-domain response

$$B_k(z) = \sum_{l=1}^{N_{\text{fff}}} b'_k z^{-l} ,$$

where  $N_{\text{fff}}$  is the number of tap coefficients in FFF 14, as defined above. FBF 20 is an FIR filter with z-domain response

$$D_k(z) = \sum_{i=1}^{N_{fbf}} d_k^i z^{-i} ,$$

where  $N_{fbf}$  is the number of tap coefficients in FBF 20, as also defined above.

For FIR filters, it is convenient to describe their operation in vector and matrix notation. Accordingly, we define the vector of FFF coefficients for adaptive DFE 12 as

$$\underline{b}'_k = [b_k^1 \quad \dots \quad b_k^M \quad b_k^{M+1} \quad \dots \quad b_k^{N_{fff}}] ,$$

where M is the main (or decision) tap of the FFF. A vector of past input samples to FFF 14 is defined as

$$\underline{r}'_k = [R_{k-1} \quad R_{k-2} \quad \dots \quad R_{k-N_{fff}}] .$$

The output of FFF 14,  $U_k$ , is thus equal to

$$U_k = \underline{b}'_k \cdot \underline{r}_k .$$

We define the vector of coefficients for FBF 20 as

$$\underline{d}'_k = \begin{bmatrix} d^1_k & d^2_k & \dots & d^{N_{fbf}}_k \end{bmatrix}$$

and a vector of past input samples to FBF 20 as

$$\underline{x}'_k = \begin{bmatrix} X_{k-1} & X_{k-2} & \dots & X_{k-N_{fbf}} \end{bmatrix}.$$

Assuming that slicer 26 makes correct decisions (i.e.,  $X_k$  is equal to  $A_{k-M}$ ), then the output of FBF 20,  $Q_k$ , is equal to

$$Q_k = \underline{d}'_k \cdot \underline{x}_k.$$

The output of adaptive equalizer 12, namely  $Y_k$ , is obtained by adding outputs of FFF 14 and FBF 20, as follows

$$Y_k = U_k + Q_k.$$

The decision error,  $E_k$ , is equal to the difference of the recovered signal  $X_k$  and the output  $Y_k$

$$E_k = X_k - Y_k.$$

In this example, the decision error is equal to the true error, since correct decisions are assumed.

The LMS tap update equation for FBF 20 is

$$\underline{d}'_k = \underline{d}'_{k-1} + \alpha \cdot E_k \cdot \underline{x}'_k,$$

where  $i$  is the tap index and  $\alpha$  is the LMS step size of FBF 20.

The LMS tap update equations for FFF 14 are:

$$\begin{aligned} b'_k &= b'_{k-1} + \beta_1 \cdot E_k \cdot R_{k-1}, & i &= 1, \dots, M \\ b'_k &= b'_{k-1} + \beta_2 \cdot E_{k-L} \cdot V_{k-1-L}, & i &= M+1, \dots, N_{\text{fff}} \end{aligned}$$

where  $i$  is the tap index and  $\beta_1$  and  $\beta_2$  are the LMS step sizes of FFF 14.

Adaptive DFE 12 generates the noise estimate  $V_k$  by subtracting a replica of the signal  $R_k$  from a delayed version of that signal plus noise. The signal replica is constructed by filtering  $X_k$  using CE 32. CE 32, as noted, is an FIR filter with the following z-domain response

$$G_k(z) = \sum_{l=1}^{N_{\text{ce}}} g'_k z^{-l},$$

where  $N_{ce}$  is the number of tap coefficients in the CE.

We define the vector of coefficients for CE 32 as

$$\underline{g}'_k = [g_k^1 \quad g_k^2 \quad \dots \quad g_k^{N_{ce}}]$$

and a vector of past input samples to CE 32 as

$$\underline{x}'_{CE_k} = [X_{k-1} \quad X_{k-2} \quad \dots \quad X_{k-N_{ce}}].$$

The output of CE 32, namely  $T_k$ , is thus equal to

$$T_k = \underline{g}'_k \cdot \underline{x}_{CE_k}.$$

The signal estimate  $T_k$  is a delayed estimate of the signal component of  $R_k$ .  $T_k$  is an estimate of the signal component of  $R_{k-M-L}$ . The estimate is delayed by  $M+L$  samples, where  $L$ , as noted above, is the number of precursor samples to be replicated by CE 32.

The output,  $V_{k-M-L}$ , of the CE summing node 38 is obtained by subtracting  $T_k$  from  $R_{k-M-L}$ , as follows

$$V_{k-M-L} = R_{k-M-L} - T_k.$$

$V_{k-M-L}$  contains both additive noise and a residual signal component.  $V_{k-M-L}$  is thus a relatively accurate estimate of the additive noise when the number of CE taps is sufficient and the CE coefficients have been adapted to the channel.

The LMS process adaptively updates the CE coefficients. The tap update equations (in vector notation) for CE 32 are

$$\underline{g}_k = \underline{g}_{k-1} + \gamma \cdot V_{k-M-L} \cdot \underline{x}_{CE_k},$$

where  $\gamma$  is the CE LMS step size.

The LMS step sizes of adaptive DFE 12 may be optimized with respect to its signal-to-noise ratio (SNR). The LMS step sizes  $\alpha$  and  $\beta_1$  are bounded by the eigenvalues of the filter and the desired tap fluctuation error. To keep the gain of taps M+1 to N<sub>fff</sub> of FFF 14, with respect to the noise, equivalent to the gain of FBF 20, with respect to the ISI, the LMS step size  $\beta_2$  may be increased by a factor equal to the SNR. For example, assume the SNR at the input to slicer 26 is 30 decibels (dB), then

$$\beta_2 = (10^{30/20}) * \beta_1.$$

Adaptive DFE 12 represents only one embodiment of the invention; other embodiments exist. For example, the use of CE 32 is only one possible way of obtaining the noise or noise estimate. When other methods of obtaining the noise or noise estimate are used, the FFF tap update equations may be slightly modified, e.g., in terms of the time indices of the decision error and the noise estimate.

The invention may also be used with a fractionally-spaced FFF. In this case, the noise is estimated at the fractional sampling rate of the FFF. This can be done by utilizing multiple CE filters, each operating at the symbol rate, or one CE filter operating at the fractional sampling rate.

The invention is described here for a real, baseband communications system, however, it is also valid for a complex or passband system. The tap update equations may be slightly modified, in this case, to account for complex arithmetic and to include conjugation of the data.

The coefficients  $b_k^i$ ,  $d_k^i$  and  $g_k^i$ , described above, may be adaptively determined using hardware, e.g., discrete hardware components such as programmable logic gates, or software running on a computer. Fig. 3 shows a computer 40 on which the coefficients may be determined. Computer 40 includes a processor 42, a memory 44, and a storage medium 46 (see view



48). Storage medium 46 stores machine-executable instructions  
50 that are executed by processor 42 out of memory 44 to  
adaptively determine the coefficients.

Although a personal computer is shown in Fig. 3, the  
invention is not limited to use with the hardware and software  
of Fig. 3. It may find applicability in any computing or  
processing environment. The coefficients may be determined  
5 using hardware, software, or a combination of the two. The  
coefficients may be determined using computer programs  
executing on programmable computers or other machines that  
each include a processor, a storage medium readable by the  
processor (including volatile and non-volatile memory and/or  
10 storage components), at least one input device, and one or  
more output devices. Program code may be applied to data  
entered using an input device (e.g., a mouse or keyboard) to  
determine the coefficients.

Each such program may be implemented in a high level  
15 procedural or object-oriented programming language to  
communicate with a computer system. However, the programs can  
be implemented in assembly or machine language. The language  
may be a compiled or an interpreted language.

Each computer program may be stored on a storage medium/  
20 article (e.g., CD-ROM, hard disk, or magnetic diskette) that

is readable by a general or special purpose programmable computer for configuring and operating the computer when the storage medium or device is read by the computer to determine the coefficients. The coefficients may be determined using a machine-readable storage medium, configured with a computer program, where, upon execution, instructions in the computer program cause a machine to determine the coefficients.

The invention is not limited to the specific embodiments described above. For example, the invention is not limited to use with FIR filters or to use with the particular configuration shown in Fig. 2. The adaptive equalizer may be implemented in a single pair high speed digital subscriber line (HDSL2/G.SHDSL) system, or any other signal transmission system that requires reduction in ISI and colored noise.

Alternatively, the coefficients  $b_k^i$  may be generated based on the input signal alone, i.e., not the noise. This is called the zeroforcing (ZF) criterion.

Other embodiments not described herein are also within the scope of the following claims.

What is claimed is: